CLAIMS

1 – A method for processing video pictures for display on a display device having a plurality of luminous elements corresponding to the pixels of a picture wherein the time of a video frame or field is divided into a plurality of N sub-fields (SF) during which the luminous elements can be activated for light emission in small pulses corresponding to a sub-field code word of n bits used for coding the p possible video levels lighting a pixel, comprising the steps of :

- determining if pictures are static pictures or moving pictures,
- in case of static pictures, processing video pictures using a first sub-field encoding method adapted to pictures when no motion is detected, and
- in case of moving pictures, processing video pictures using a second encoding method reducing dynamic false contour effect adapted to pictures when motion is detected.

2 – A method according to claim 1, characterized in that the first sub-field encoding method is a standard encoding method wherein the n bit video level is decomposed in n or more lighting sub-periods or sub-fields.

3 – A method according to claim 1, characterized in that the second sub-field encoding method is a method wherein among the set of p possible video levels for lighting a pixel, a sub-set of m video levels with $n < m < p$ is selected, which is used for light generation, wherein the m values are selected according to the rule that the temporal centre of gravity for the light generation of the corresponding sub-field code words grow continuously with the video level apart in the low video level range up to a first predefined limit and/or in the high video level range from a second predefined limit on.

4 – A method according to claim 3, characterized in that the temporal centre of gravity for the light generation is defined according to the following formula :

$$CG(code) = \frac{\sum_{i=1}^{n} sfW_i * \delta_i(code) * sfCG_i}{\sum_{i=1}^{n} sfW_i * \delta_i(code)}$$

there sfW_i is the sub-field weight of the i^{th} sub-field, δ_i is equal to 1 if the i^{th} sub-field is activated and zero if the i^{th} sub-field is inactivated and $sfCG_i$ is the temporal centre of gravity for the light generation of the i^{th} sub-field.

5 – A method according to claims 3 or 4, characterized in that, in case of a sub-field organization characterised by a specific number of sub-fields with a specific series of sub-field weights for a colour component value, more than one corresponding sub-field code word exists, the set of possible sub-field code words is reduced by taking only those sub-field code words for each video level, which have the minimum binary value for the case that in a sub-field code word the weight of each bit is ordered according to size.

6 – A method according to claim 5, characterized in that the selection of video levels from the further reduced set of sub-field code words is performed by taking only one video level on each group of sub-field code words having the same radical on the MSB side, namely the video level belonging to the next higher group of sub-field code words and having the smallest centre of gravity superior to the centre of gravity of the previous selected video level, wherein in case that the next higher group of sub-field code words does not provide a sub-field code word having a centre of gravity inferior to the previous one, then the second next higher sub-field code word group will be chosen for selecting the next video level and so on.

7 – A method according to one of the claims 1 to 6, characterized in that the method comprises an additional step of determining if static and moving pictures are noisy pictures or free noise pictures and using an

adaptive noise dithering for noisy pictures or a noise free dithering for noise free pictures.

8 – A method according to claim 7, characterized in that the adaptive noise dithering is an error diffusion algorithm.

9 – A method according to claim 7, characterized in that the noise free dithering is a 3D dithering.

10 – A method according to one of the claims 1 to 9, characterized in that, when information about motion is available, the decision to switch between the first sub-field encoding method and the second sub-field encoding method in view of the detection of the motion is done according to the following algorithm :

- for each frame,

if MOTION is ON then MOTION-counter = 2 x MOTION-LIMIT,

if MOTION is OFF then MOTION counter = MOTION counter – 1.

When MOTION-counter < MOTION-LIMIT then the first coding sub-field method is activated else the second coding sub-field method is maintained, the Motion-counter being a counter able to count from 0 to 2 x MOTION-LIMIT, if MOTION-counter< 0 then MOTION-counter = 0.

11 – A method according to one of the claims 1 to 9, characterized in that, when information about noise is available, the decision to switch between an adaptive noise dithering and a noise free dithering is done according to the following algorithm :

For each frame,

If NOISE is ON then NOISE-counter = NOISE-counter+1.

If NOISE is OFF, then NOISE-counter = NOISE-counter-1.

When NOISE-counter > NOISE-LIMIT, an active adapted noise dithering is activated else a noise free dithering is used,

the NOISE-counter being a counter able to count from 0 to 2x NOISE-limit, if NOISE-counter < 0 then NOISE-counter = 0 and if NOISE-counter > 2 x NOISE-LIMIT then NOISE-counter = 2 x NOISE-LIMIT.

12 – A method according to one of claims 1 to 9, characterized in that, when no information about motion is available, the motion detection is done by using the following method comprising the steps of :

- splitting picture in M detection zones Z(i,j);
- computing for each detection zone Z(i,j) an histogram of the zone content,
- for each zone, comparing the computed histogram with the corresponding histogram of the previous picture,
- if the difference is above a given limit, then putting Motion ON.

13 – A method according to claim 12, characterized in that the histogram of the zone is defined by choosing a number of discrete regions, each one defined by lower and upper video frontiers.

14 – A method according to claims 12 and 13, characterized in that, the determination of motion in picture is done by the following algorithm :

```
DifferenceZone=0
For each zone Z(i;j,n)
{
   DifferenceRegion=0
   For each region R[k,n]
   {
   If (|R[k,n] - R[k,n-1]| > RegionLimit) then DifferenceRegion++
   }
   If (DifferenceRegion > ZoneLimit) then DifferenceZone++
}
```

If (DifferenceZone>FrameLimit) then Motion=ON else Motion=OFF.

15 – A method according to one of claims 1 to 9, characterized in that, when no information about noise is available, the noise detection is done with a method comprising the following steps of :

- dividing the picture in segments L constituted by a set of N pixels taken from two consecutive lines,

- calculating a noise estimation for the segment L :

$$Noise(L) = \frac{1}{N} \sum_{i=1}^{i=N} (A(i) - B(i))$$

- defining the noise estimation for the whole picture as the minimum value of the noise estimation Noise = $\min_L$ (Noise (L)),

and comparing the value Noise to a threshold to determine if Noise is ON or OFF.

16 – A method according to one of claims 1 to 9, characterized in that, when no information about noise is available, the noise detection is done with a method :

- defining in blanking area of a picture, n' regions Reg(n') made of m pixels x l lines,
- computing the mean values of each region with the following formula :

$$\text{MGL}(n') = \frac{1}{m\ x\ l} \sum_{i,j} Pict(i;j) \text{ where}$$

i and j represent the various horizontal and vertical pixel positions of each region :

- computing, for each region, the mean square error :

$$\text{MSE}(n') = \frac{1}{m\ x\ l} \sum_{i,j} (Pict(i;j) - \text{MGL}(n^1))^2\text{, then}$$

- estimating the final noise of the whole picture.

17 – A method according to claim 16, characterized in that, the estimation of the final noise is done using a histogram representation of the various mean square errors MSE(n'), the minimum value in this histogram being defined by the minimal error located in a digital picture given by $MSE_{\min} = \frac{1}{12}$ since $MSE_{\min} = \int_{-1/2}^{1/2} (x^2)dx = \left[\frac{x^3}{3}\right]_{-1/2}^{1/2} = \frac{1}{3} \cdot \left[\frac{1}{8} - \frac{1}{-8}\right]$ where x^2 represents the square error integrated among the error dynamic; the error dynamic being [-1/2;1/2] with a quantification step of 1.

On the horizontal axis of the histogram, various segment are chosen representing a domain of the value MSEx12.

In the vertical axis, the occurrence of each domain is computed as following :

For all region n

{

For all domain k

{

if (Min(k) < MSE(n) ≤ Max(k)) then Histo[k]++

}

}

Then, the highest value in the occurrence of the histogram is chosen to quantify the MSE of the whole picture.

- Depending on a noise limit, if MSE > NOISE-limit then NOISE = ON else NOISE = OFF.

18 – A method according to one of claims 1 to 9, characterized in that, when no information about noise is available, the noise detection is done with a method comprising the following steps :

- dividing each picture in small blocks Block(n),
- for each block, performing an estimation of the mean square error :

$$\mathrm{MSE(n)} = \frac{1}{N} \sum_{Block(n)} (Pict(i,j,t) - Pict(i,j,t-1))^2$$

Where t represents the current frame and t-1 the previous frame, i and j represent the horizontal and vertical positions of each pixels :

- defining the noise level as the smallest MSE(n) for all blocks Block(n),
- comparing the various MSE(n) to a value MOTION BLOCK,
- if MSE(n) ≥ MOTION BLOCK, MOTION counter is increased, the MOTION-counter being initialised at the beginning of each frame and,

- when MOTION counter $\geq$ MOTION -limit then MOTION = ON else MOTION = OFF.

19 – A method according to one of claims 1 to 9, characterized in that, when no information about noise is available, the noise detection is done with a method comprising the following steps of :

- dividing the current frame (t) in small blocks Block(n)
- computing, for each corresponding block of a searching area in the previous frame (t-1) a mean square error :

$$MSE(\delta x,\delta y,n) = \frac{1}{N} \sum_{Block(n)} (Pict(i,j,t) - Pict(i-\delta x,j-\delta y,t-1))^2$$

Where N represents the size of the block n,

- computing, for each position in the searching area ($\delta x,\delta y$), the smallest MSE for a block n,
- choosing, $MSE_{min}(n) = \min_{\delta x,\delta y}(MSE(\delta x,\delta y)$ for noise estimation,
- choosing Motion(n)=1 if $MSE_{min}(n)$ is given for $(\delta x,\delta y)=(0,0)$ else Motion(n)=0
- computing, using the various $MSE_{min}(n)$ an histogram of errors for the noise estimation,
- computing a value Motion-sum = $\sum_{all\ block\ n} Motion(n)$,

comparing this value to a limit and if Motion-sum $\geq$ Motion -limit then MOTION = ON else MOTION = OFF.

20 – A system for implementing the method according to claims 1 to 19, characterized in that it comprises a front end IE (10) delivering information about noise and motion, a dedicated plasma IC (11) and a plasma panel 12.

21 – A system for implementing the method according to claims 1 to 9 and 12 to 18, characterized in that it comprises a front end IC (10') delivering no information about noise and motion, a dedicated plasma IC (11') incorporating specific methods for detecting noise and motion, a microcontroller (13') and a plasma panel (12').

22 – A system according to claim 21, characterized in that it further comprises a frame memory (14') to implement the method according to claim 19.